A Statistical Performance Analysis of Graph Clustering Algorithms

A Statistical Performance Analysis of Graph Clustering Algorithms

Pierre Miasnikof[1(✉)], Alexander Y. Shestopaloff[2], Anthony J. Bonner[3], and Yuri Lawryshyn[1]

[1] Department of Chemical Engineering and Applied Chemistry, University of Toronto, Toronto, Canada
p.miasnikof@mail.utoronto.ca
[2] The Alan Turing Institute, London, UK
[3] Department of Computer Science, University of Toronto, Toronto, Canada

Abstract. Measuring graph clustering quality remains an open problem. Here, we introduce three statistical measures to address the problem. We empirically explore their behavior under a number of stress test scenarios and compare it to the commonly used modularity and conductance. Our measures are robust, immune to resolution limit, easy to intuitively interpret and also have a formal statistical interpretation. Our empirical stress test results confirm that our measures compare favorably to the established ones. In particular, they are shown to be more responsive to graph structure, less sensitive to sample size and breakdowns during numerical implementation and less sensitive to uncertainty in connectivity. These features are especially important in the context of larger data sets or when the data may contain errors in the connectivity patterns.

1 Introduction

While there are many graph clustering[1] algorithms in the literature (e.g., [15,17,21,24]), measuring their performance, that is assessing the quality of the clusters they identify, remains an open problem [1,3,6,11–13,16,23]. Graph clustering is a form of unsupervised learning, where one typically cannot count on labeled data to assess results. For example, in [20], the authors correctly assert that "*(...) running a clustering algorithm over a set of randomly generated data points will always produce clusters which, however, have little meaning.*" Therefore, our only quality measure is a thorough assessment of the graph's and resulting clusters' connectivity patterns.

In this article, we present new clustering performance measures to assess the strength of the clustering returned by a specific algorithm and compare clusterings across algorithms on a specific graph. We restrict our attention to undirected

[1] Note on vocabulary: Although there are subtle differences between the concepts of graph clustering and community detection, in this document we use the two interchangeably.

A. Bonato et al. (Eds.): WAW 2018, LNCS 10836, pp. 170–184, 2018.
https://doi.org/10.1007/978-3-319-92871-5_11

unweighted and weighted graphs, with no self-loops or multiple edges. We begin with a review of two of the most common clustering performance measures, modularity and conductance. We empirically demonstrate how these measures may, in some cases, be drowned out by graph structure and lack sensitivity. We also offer three alternative measures, which are shown to be more robust.

2 Performance Measures

In this section, we describe the two most popular performance metrics in the literature, namely modularity and conductance. We also present our own statistical measures, the "Kappas". In the following sections, we will empirically analyze their strengths and weaknesses.

2.1 Modularity

Modularity (Q) is by far the most popular measure of clustering performance [4,5,8,13,17–19]. It was originally introduced by Newman and Girvan in 2004 [17] and has been extensively used both as a performance measure and objective function for clustering algorithms (e.g., [2,7,17]). In this section, we present modularity (Q) as defined in [5].

$$Q = \sum_{i=1}^{k} \left(\underbrace{e_{i,i} - a_i^2}_{q_i} \right)$$

Where,

$$e_{i,i} = \frac{1}{2m} \sum_{v,w} A_{v,w}\, \delta(c_v, i)\delta(c_w, i)$$

$$a_i = \frac{1}{2m} \sum_{v} A_{v,\cdot}\, \delta(c_v, i)$$

Here, $m = |E|$ is the total number of edges in the graph, k is the number of clusters, $A_{v,w}$ is the element at the intersection of the v-th row and w-th column of the adjacency matrix, A_v, is the entire v-th row of the adjacency matrix, $\delta(x, y)$ is the Kroenecker delta function, $e_{i,i}$ is the portion of vertex degree connecting vertices within cluster i, a_i is the total vertex degree in cluster i and c_v is the cluster in which vertex 'v' is clustered into by the algorithm. Putting it together, we get

$$Q = \sum_{i=1}^{k} \left[\underbrace{\frac{1}{2m} \sum_{v,w} A_{v,w}\, \delta(c_v, i)\delta(c_w, i)}_{e_{i,i}} - \underbrace{\frac{1}{4m^2} \left(\sum_{v} A_{v,\cdot}\, \delta(c_v, i) \right)^2}_{a_i^2} \right]. \quad (1)$$

(A high value indicates densely connected clusters.)

In closing, it should be noted that modularity suffers from a resolution limit, as described by Fortunato and Bathélemy [9]. These authors describe how any (clustering) quality function that is defined as a sum of qualities of individual clusters where terms from smaller clusters are dominated by terms from larger clusters suffers from resolution limit. Because the smaller clusters' contribution to the sum is dominated by the larger clusters, the final result is also dominated and does not always reflect the structure accurately. Indeed, in (1), we see how larger clusters dominate the outer summation.

2.2 Conductance

Conductance (ϕ, Φ) is another popular clustering performance measure [6,13,14, 22,23]. In this article, we use the definition presented in [22].

At the individual cluster level,

$$\phi(S) = \frac{\partial(S)}{\min\left(d(S), d(V \setminus S)\right)}$$

At the graph level,

$$\Phi(G) = \min_{S} \phi(S)$$

Here, $\partial(S)$ is the number of edges joining vertices in cluster S to vertices outside S, $d(S)$ is the sum of vertex degrees within S and $d(V \setminus S)$ the sum of vertex degrees on the graph, outside S. (A low conductance indicates strongly connected clusters.)

2.3 The Kappas

Our overarching goal in developing our measures is to gauge the strength of connectivity on the graph in general, within individual clusters and between clusters. While the established measures of clustering strength, modularity and conductance, measure intra-cluster connectivity strength, we seek to measure the strength of intra- and inter-cluster connectivity relative to the overall graph's connectivity. For example, in a densely connected graph we expect clusters to be even more strongly connected and strong inter-cluster connections can be consistent with a good partition. Conversely, in a densely connected graph, poorly connected clusters or strong inter-cluster connectivity are symptoms of a poor clustering.

We define $\bar{K}$ as the graph's overall connectivity ratio, $\bar{K}_{\text{intra}}$ as the measure of intra-cluster connectivity and $\bar{K}_{\text{inter}}$ as the measure of inter-cluster connectivity. According to every definition of a good clustering, we expect that an efficient clustering algorithm will label vertices such that intra-cluster connectivity is greater than inter-cluster connectivity [8,18,19] (if the graph does indeed have a clustered structure). Under our model, we expect that a good clustering will group vertices so they form clusters whose vertices are more densely connected

than the average connection between any two vertices on the graph. Similarly, we expect that a good clustering will group vertices so they form clusters whose vertices are less densely connected to those in other clusters than the average connection between any two vertices on the graph. In summary, we expect that under a good clustering the inequalities $\bar{K}_{\text{intra}} > \bar{K} > \bar{K}_{\text{inter}}$ will hold. Our model also allows these inequalities to be formulated as a hypothesis test, as will be shown later.

Below, we present the formulation for our clustering measures, for an unweighted undirected graph, but our metrics easily generalize to weighted graphs as well. In our formulation, we use the following variables: The set of all clusters is $C = \{C_1, \ldots, C_k\}$, with $|C| = k$, the total number of vertices in the graph is N, the total number of vertices in cluster i is $|C_i| = n_i$, the set of all edges on the graph is $E = \{e_1, \ldots, e_m\}$, where $|E| = m$. Finally, $E_{i,j}$ is the set of edges connecting a vertex in cluster i to a vertex in cluster j, and $|E_{i,j}| = m_{i,j}$. As a special case, note that $E_{i,i}$ is the set of edges within cluster i, and $m_{i,i}$ is the number of edges connecting vertices within cluster i.

In order to gauge the strength of the entire graph's, of each cluster's and each inter-cluster pair's connectivity, we take the ratio of the observed edges over the maximum possible number of edges given the number of vertices. For inter and intra cluster connectivity, we compute the ratio for each cluster or pair of clusters and take their mean as a graph-wide measure. All our measures lie in the $[0, 1]$ interval, with high values denoting highly connected graphs, clusters or cluster pairs and vice-versa.

We define the graph's connections ratio as

$$\bar{K} = \frac{|E|}{0.5 \times N(N-1)}.$$

The graph's connection ratio is the ratio of the total number of edges over the number of edges in a complete graph with the same number of vertices. The closer $\bar{K}$ is to 1, the closer the graph is to being a complete graph. Conversely, the closer $\bar{K}$ is to 0, the closer the graph is to being a set of disconnected vertices.

We also define the mean intra-cluster connections ratio as

$$\bar{K}_{\text{intra}} = \frac{1}{K} \sum_{i=1}^{\kappa} \frac{|E_{i,i}|}{0.5 \times n_i(n_i - 1)}.$$

The mean intra-cluster connections ratio is the mean ratio of the number of edges within each cluster over the maximum number of edges that could possibly connect the vertices of each cluster. Each term in the summation is a measure of how closely each cluster is to being a clique. Each always lies on the interval $[0, 1]$, with a value of 0 indicating a cluster is just a set of disconnected vertices and a value of 1 indicating that a cluster is a clique. At the aggregate level, $\bar{K}_{\text{intra}}$ is the sample mean of the individual terms and also lies in the interval $[0, 1]$. Values close to 0 indicate poorly connected clusters on average, while values closer to 1 indicate densely connected clusters on average.

Finally, we define the mean inter-cluster connections ratio as

$$\bar{K}_{\text{inter}} = \frac{1}{0.5 \times k(k-1)} \sum_{i=1}^{k} \sum_{j=i+1}^{k} \frac{|E_{i,j}|}{0.5 \times ((n_i + n_j)(n_i + n_j - 1) - n_i(n_i - 1) - n_j(n_j - 1))}.$$

The mean inter-cluster connections ratio is the mean ratio of the number of edges joining vertices in two different clusters, over the total number of edges that could possibly connect each pair of vertices in each cluster pair (c_i, c_j). Each term in the double summation is a measure of how closely two clusters 'i' and 'j' are from forming a single clique. Each of these terms also lies in the interval $[0, 1]$, with a value of 0 indicating no connection between a pair of clusters and a value of 1 indicating that the pair of clusters forms a clique. At the aggregate level, $\bar{K}_{\text{inter}}$ is the sample mean of the individual terms of the summation and also lies in the interval $[0, 1]$. Values close to 0 indicate poor inter-cluster connections, on average, a desirable feature indicating strong cluster partitions, On the other hand, values closer to 1 indicate improperly partitioned clusters, on average.

It should also be mentioned that in cases where the connectivity patterns of the clusters is very noisy, the median of the summation terms can be used in lieu of the mean, in order to produce more robust measures. Unfortunately, this substitution makes statistical interpretation and significance testing less obvious.

Resolution Limit and Sensitivity to Cluster Size. It is important to note that neither $\bar{K}_{\text{intra}}$ nor $\bar{K}_{\text{inter}}$ are affected by individual cluster size and do not suffer from the resolution limit observed in modularity [9]. Large clusters do not skew their values, since all terms in the sums are scaled by the total number of possible edges within each cluster or pair of clusters and always lie on the $[0, 1]$ interval. This feature makes these measures robust to large "mega-clusters" that are often observed in real-world networks and to the fallacious tendency of clustering algorithms to lump all vertices together in a few very large clusters. (Naturally, $\bar{K}$ is a graph-wide measure that remains completely agnostic to clusters and their respective sizes.)

The equal weight carried by each cluster or pair of cluster does, however, have its drawbacks. Because our measures are unweighted means, they are somewhat sensitive to outliers. For example, a few unrepresentative small clusters could indeed skew the measures. However, the effect of outliers is typically smoothed out by the mean or can be corrected by the use of a weighted mean.

Statistical Interpretation of the Kappas. The main strength of our Kappas comes from their statistical definition. In the unweighted case, $\bar{K}$ is the probability any two nodes are connected, and in the weighted case it becomes the mean edge weight. Similarly, $\bar{K}_{\text{intra}}$ ($\bar{K}_{\text{inter}}$) is the mean probability two nodes within a cluster (between clusters) are connected or the mean intra-cluster (inter-cluster) edge weight.

In probabilistic terms, we expect a good clustering to partition the graph such that the probability there exists an edge $(e_{i,j})$ between two arbitrary nodes

'i' and 'j' is lower than the probability a connection exists if these nodes are in the same cluster (i.e., if $c_i = c_j$) and higher than when they belong to different clusters (i.e., $c_i \neq c_j$). Mathematically, we expect the following to hold:

$$Pr\left[e_{i,j}|c_i = c_j\right] > Pr[e_{i,j}] > Pr[e_{i,j}|c_i \neq c_j]$$

In the case of a weighted graph, these probabilities become expected values of edge weights between arbitrary vertices, vertices within and vertices between clusters, and we expect the following inequalities to hold:

$$E[e_{i,j}|c_i = c_j] > E[e_{i,j}] > E[e_{i,j}|c_i \neq c_j]$$

Defining our measures in this way, as estimates of an unknown "true" parameter, with an associated standard error, allows formal significance testing using a simple t-test. Such tests can be used to determine if the clusters identified by an algorithm are statistically significant. If they are, we expect the inequalities $\bar{K}_{\text{intra}} > \bar{K} > \bar{K}_{\text{inter}}$ to hold at a reasonable significance level. These inequalities are necessary and sufficient to conclude the clusterings returned by an algorithm are statistically (on average) consistent with the universally accepted definition of a clustering. [8,18,19]. Our statistics can also be used when comparing two or more algorithms' performances on a given graph. In such a case, in order to conclude algorithm 'a' is better than algorithm 'b', we should observe $\bar{K}^a_{\text{intra}} > \bar{K}^b_{\text{intra}}$ and $\bar{K}^a_{\text{inter}} < \bar{K}^b_{\text{inter}}$.

Finally, let us note that our statistical (i.e., non deterministic) definition also allows for uncertainty in the connectivity, another open problem [10]. Unlike modularity and conductance, our measures are defined as statistical measurements with associated standard errors, not deterministic quantities.

To formally confirm statistical significance and the strength with which the sufficient conditions are met, we formulate an appropriate null hypothesis and apply the t-test. Examples of such a test are shown in Sect. 4.4.

3 Computational Experiments

In order to empirically assess the accuracy of the various performance measures, to study their response to various graph structures and cluster labelings, we subject them to a number of numerical stress test scenarios, using simulated graphs and labels. The full experimental set-up of our individual tests and scenario details are described in the next sub-section.

Overall, our goal is to test the accuracy and robustness of our clustering measures and compare their behavior to that of the two main clustering measures in the literature (modularity and conductance). Simulation is used to generate test scenarios where the clustering structure is known in advance and could be modified easily. These test scenarios are then used to examine and compare the sensitivities of the kappas, modularity and conductance. Our scenarios include a number of contrived instances, which are useful to stress test our metrics through "extreme" examples and compare their behavior to those of the more established measures.

The overarching logic guiding our tests is that a good measure of inter- or intra- cluster connectivity should accurately reflect the simulated graph's structures. We would expect measures of intra-cluster connectivity, K_{intra} and modularity to increase in step with the simulated graph's connectivity levels, while we would expect conductance to display the inverse behavior. We would also expect K_{inter} to follow the fluctuations of inter-cluster connectivity.

It should also be mentioned that some authors have used so-called "ground-truth" data sets, networks where the nodes' cluster memberships were labeled, as benchmarks for clustering algorithm performance (e.g., [16,23,24]). Our approach is more general, data set and objective function independent. Arguably, the fact that an algorithm anecdotally provided accurate clustering on a labeled instance is no guarantee it will perform equally well on another (likely unlabeled) instance. In addition, our experiments provide us with an understanding of each measure's sensitivity and response to graph structure.

3.1 Experimental Set-Up

In the first set of experiments, shown in Table 1, we examine the effect of intra-cluster connectivity. We begin with a graph with no edges between any of the vertices and gradually increase intra-cluster connectivity in steps of 25%, while maintaining inter-cluster connectivity at 0% (e.g., 25% of nodes are connected to another node within their assigned cluster, 75% of nodes in each cluster have no connections at all, nodes with connections only have connections to other nodes within their assigned cluster, each cluster is disconnected from the rest of the graph).

We then examine the effect of inter-cluster connectivity on each measure. We begin with no inter-cluster connectivity and then increase it in steps of 25% (e.g., 25% of nodes are connected to 25% of nodes outside their cluster), while keeping intra-cluster connectivity at 0%. In other words, clusters are just sets of disconnected vertices. In these scenarios, we imagine an algorithm, one with a very poor cluster detection ability, that groups disconnected vertices into clusters with different levels of inter-connection to other clusters but with an intra-cluster connectivity that remains constant at 0%. Results are shown in Table 2.

In our experiments, we expect $\bar{K}_{\text{intra}}$ to increase in step with intra-cluster connection percentage. We also expect $\bar{K}_{\text{inter}}$ to increase in step with inter-cluster connection percentage. If this in-step increase occurs, it indicates our measures accurately detect the graph's connectivity structure.

Finally, in order to assess our measures' robustness, we repeat all the tests described above, but with the introduction of "noise" in the connectivity patterns. Noise is introduced in the form of 100% intra-(inter-) cluster connectivity. Results are shown in Tables 3 and 4.

In the tables that follow, we also report each graph's characteristics, for each experiment. The total number of vertices is denoted by N, the total number of clusters by $|C|$, and the total number of edges by $|E|$.

Table 1. Varying intra-cluster connectivity, no noise from inter-cluster connectivity

Pct Inter = 0, Pct Intra varies					
Pct Intra	0	25	50	75	100
N	10,048	9,440	9,666	10,493	10,039
$\|C\|$	200	200	200	200	200
$\|E\|$	0	76,942	160,147	269,341	336,942
$\bar{K}$	0.00	0.00	0.00	0.00	0.01
$\bar{K}_{intra}$	0.00	0.26	0.50	0.75	0.99
Std Err ($\bar{K}_{intra}$)	0.00	0.01	0.01	0.01	0.01
$\bar{K}_{inter}$	0.00	0.00	0.00	0.00	0.00
Std Err ($\bar{K}_{inter}$)	0.00	0.00	0.00	0.00	0.00
Φ	0.00	0.00	0.00	0.00	0.00
Q	0.00	0.99	0.99	0.99	0.99

Table 2. Varying inter-cluster connectivity, no noise from intra-cluster connectivity

Pct Intra = 0, Pct Inter varies					
Pct Inter	0	25	50	75	100
N	10,530	10,089	9,354	10,028	10,829
$\|C\|$	200	200	200	200	200
$\|E\|$	0	3,058,924	10,753,463	27,815,367	58,250,108
$\bar{K}$	0.00	0.06	0.25	0.55	0.99
$\bar{K}_{intra}$	0.00	0.00	0.00	0.00	0.00
Std Err ($\bar{K}_{intra}$)	0.00	0.00	0.00	0.00	0.00
$\bar{K}_{inter}$	0.00	0.06	0.24	0.55	1.00
Std Err ($\bar{K}_{inter}$)	0.00	0.00	0.00	0.00	0.00
Φ	0.00	1.00	1.00	1.00	1.00
Q	0.00	−0.01	−0.01	−0.01	−0.01

4 Discussion

As shown in Sect. 3, our "Kappas" behave as expected, even when subjected to noise. In all instances where the labeling of clusters reflects a good partition, the inequalities $\bar{K}_{\text{intra}} > \bar{K} > \bar{K}_{\text{inter}}$ hold and they do not hold in instances where the partition reflects poor clustering. For example, in Table 3, all instances are cases of poor clustering, by design. Similarly, in Table 4, instances where the percentage of inter-cluster connectivity is below 75% are examples designed to show good clustering and our inequalities hold in each.

More importantly, our inter- and intra-cluster measures follow the fluctuations of the graph's connectivity patterns more accurately than either modularity

Table 3. Varying intra-cluster connectivity, with noise from inter-cluster connectivity

Pct Inter = 100, Pct Intra varies					
Pct Intra	0	25	50	75	100
N	10,048	10,096	10,526	10,115	10,182
$\|C\|$	200	200	200	200	200
$\|E\|$	50,142,540	50,712,690	55,215,342	51,067,113	51,831,471
$\bar{K}$	0.99	1.00	1.00	1.00	1.00
$\bar{K}_{intra}$	0.00	0.25	0.50	0.74	0.98
Std Err (intra)	0.00	0.00	0.01	0.01	0.01
$\bar{K}_{inter}$	1.00	1.00	1.00	1.00	1.00
Std Err (inter)	0.00	0.00	0.00	0.00	0.00
Φ	1.00	1.00	1.00	0.99	0.99
Q	−0.01	0.00	0.00	0.00	0.00

Table 4. Varying inter-cluster connectivity, with noise from intra-cluster connectivity

Pct Intra = 100, Pct Inter varies					
Pct Inter	0	25	50	75	100
N	9,917	9,662	10,512	10,043	10,151
$\|C\|$	200	200	200	200	200
$\|E\|$	314,102	3,127,922	13,942,175	28,187,302	51,516,325
$\bar{K}$	0.01	0.07	0.25	0.56	1.00
$\bar{K}_{intra}$	1.00	0.99	1.00	1.00	1.00
Std Err (intra)	0.00	0.01	0.00	0.00	0.00
$\bar{K}_{inter}$	0.00	0.06	0.24	0.54	1.00
Std Err (inter)	0.00	0.00	0.00	0.00	0.00
Φ	0.00	0.85	0.96	0.98	0.99
Q	0.99	0.09	0.02	0.01	0.00

or conductance. It should be noted however, that $\bar{K}_{\text{inter}}$ is less responsive to increases in inter-cluster connectivity than $\bar{K}_{\text{intra}}$ is to increases in intra-cluster connectivity and that a graph's overall connectivity ($\bar{K}$) closely reflects inter-cluster connectivity, especially in cases where the number of clusters is large. Additionally, we note modularity and conductance display very counterintuitive behaviors, although on a much larger scale. In the following sections we attempt to explain these unintuitive behaviors and explain why the "Kappas" provide a more accurate picture of the graph's and clusters' connectivity patterns than either modularity or conductance.

Finally, in the following sections, we also show that the erratic behavior displayed by modularity and conductance are the result of their sensitivity to

numerical implementation and sample sizes. This numerical sensitivity deeply affected our results with our moderately-sized graphs and clusters. As we will show in the next sections, this numerical sensitivity would only be compounded in the case of a larger data set, rendering these measures even less responsive. These sensitivities to data set size are particularly relevant in the context of large data sets ("big data").

4.1 Modularity Under Stress Test

In order to illustrate the lack of responsiveness of modularity and explain the results in the previous section, we examine the following numerical example: $|C| = 200, N = 16,400$ and $n_i = 82\ \forall i$. We then adjust the intra and inter-cluster connectivities, to examine the effect on modularity. The results are shown in Tables 5 and 6.

We begin with a clustering algorithm that would be very deficient and returns "clusters" that have 0% connection within themselves but are fully connected to the rest of the graph ($A0$). We gradually increase intra-cluster connectivity to 25% ($A25$) and 100% ($A100$), while keeping inter-cluster connectivity constant at 100%. We then do the opposite, we begin with 200 isolated complete graphs (in $B0$, each cluster is an isolated complete graph) and then increase inter-cluster connectivity to 25% ($B25$). These experiments are almost the same as those shown in Sect. 3, except that we kept cluster size constant, at 82 vertices, in order to facilitate calculations.

Table 5. Varying intra-cluster connectivity

Scenarios	A0		A25		A100	
Components of Q	e_ii	a_i	e_ii	a_i	e_ii	a_i
cluster 1	0	0.005	0.00001	0.005	0.00002	0.005
cluster 2	0	0.005	0.00001	0.005	0.00002	0.005
⋮	⋮	⋮	⋮	⋮	⋮	⋮
cluster K	0	0.005	0.00001	0.005	0.00002	0.005

In Table 5, we see that with NO connectivity within clusters, $Q \approx 200 \times (0 - 0.005^2) \approx 0$. Now if we raise the intra cluster connectivity from 0% to 25%, we add $\lceil 0.25 \times 82 \times 81 \rceil = 831$ edges to the graph, all of which connect vertices within clusters.

The a_i portion remains essentially unaffected, because the a_i of each node is scaled by $\frac{1}{4m^2}$ (i.e., increase of $\frac{831}{4m^2}$). On the other hand, $e_{i,i}$, which is scaled by $\frac{1}{2m}$ (i.e., increase of $\frac{831}{2m}$) goes up ever so slightly, but on a different order of magnitude, and the denominator (m) also increases. So in the end, the added connectivity only has an infinitesimal effect on the value of Q:

$$Q \approx 200 \times (0.00001 - 0.005^2) \approx 0$$

Increasing the intra-cluster connectivity even further to 100% does not affect the value of Q either. Indeed, the number of intra-cluster edges increases to $82 \times 81 \times 0.5 = 3,321$, but this increase is scaled by $\frac{1}{2m}$ or $\frac{1}{4m^2}$, while m also increases as well. So in the end, Q remains indistinguishable from 0, $Q \approx 200 \times (0.00002 - 0.005^2) \approx 0$.

Table 6. Varying inter-cluster connectivity

Scenarios	BO		B25	
Components of Q	e_ii	a_i	e_ii	a_i
cluster 1	0.005	0.005	0.00038	0.005
cluster 2	0.005	0.005	0.00038	0.005
⋮	⋮	⋮	⋮	⋮
cluster K	0.005	0.005	0.00038	0.005

In Table 6, we observe that when none of the vertices within clusters are connected to vertices outside their cluster, yet all have connections to vertices within their assigned clusters (case of K isolated complete graphs), $e_{i,i} = a_i$. As a result $Q \approx 200 \times (0.005 - 0.005^2) \approx 1$. But as soon as inter-cluster connectivity increases, Q collapses. Increasing inter-cluster connectivity dramatically increases m, which dramatically reduces $e_{i,i}$. Simultaneously, a_i increases, although very modestly. With 200 connected components, modularity quickly reaches its maximum, $Q \approx 200 \times (0.005 - 0.005^2) \approx 1$. With 25% inter-cluster connectivity, it quickly approaches 0, $Q \approx 200 \times (0.00038 - 0.005^2) \approx 0.07$. Note that although the degree of each vertex does indeed increase and contribute to increasing each a_i, the denominator of each a_i is $4m^2$, a graph-wide number. In the end, any increase in the cluster-centric numerator of a_i is eliminated by a dramatic graph-wide increase in m. Also note that, predictably, increases in inter-cluster connectivity beyond 25% make Q rapidly converge to zero.

4.2 Conductance Under Stress Test

Conductance is calculated at the cluster level and we assign $\Phi(G)$ the minimum value of all $\phi(S)$. Taking the minimum makes conductance very sensitive to outliers and not robust at all. In the event the graph has even one single cluster, call it $\tilde{S}$, that is densely connected, then $\phi(\tilde{S}) \approx 0$. Consequently, $\Phi(G) \approx 0$, regardless of network configuration.

In the results shown in Sect. 3, conductance breaks down for a different reason, however: In the case of an edge-less graph the denominator of conductance is zero, so we set $\phi(S) = 0$, by convention. Later, as we raise intra-cluster connectivity, the denominator remains zero, because inter-cluster connectivity is kept at 0% (Table 1). In the case of completely disconnected "clusters" (incorrectly labeled as clusters by the algorithm), the denominator is again 0. The denominator remains unchanged, when we increase inter-cluster connectivity (Table 2). This pattern repeats with the introduction of noise (Tables 3 and 4).

4.3 Kappas Under Stress Test

As shown in Sect. 3, our Kappas behave as expected, even if $\bar{K}_{inter}$ appears less responsive to graph structure than $\bar{K}_{intra}$, $\bar{K}_{inter}$ closely mirrors $\bar{K}$ and $\bar{K}_{intra}$ increases slowly in the case of our weighted examples. This relatively slow response and mirroring are completely consistent with the definitions. Note that when one edge is added anywhere on the graph, $\bar{K}$ goes up by $1/(0.5 \times N \times (N-1))$, a very small amount. When one edge is added within a cluster, $\bar{K}_{\text{intra}}$ also goes up, but by a larger amount:

$$(1/k)/(0.5 \times n_i \times (n_i - 1))$$

When an edge is added between clusters, $\bar{K}_{\text{inter}}$ also only goes up by a small amount:

$$\frac{\frac{1}{0.5 \times \kappa \times (\kappa - 1)}}{0.5 \times [(n_i + n_j)(n_i + n_j - 1) - n_i(n_1 - 1) - n_j(n_j - 1)]}$$

In the case of weighted graphs, our weights ($w_{i,j}$) are all in the $[0, 1]$ interval, so when one edge is added within a cluster $\bar{K}_{\text{intra}}$ increases by

$$(w_{i,j}/k)/(0.5 \times n_i \times (n_i - 1)) \leq (1/k)/(0.5 \times n_i \times (n_i - 1)).$$

It is also important to note that even in instances where $\bar{K}_{inter}$ or $\bar{K}_{intra}$ are not as responsive as expected, the relative magnitude of the measures still correctly identifies highly clustered graphs. In all our experiments strong clusters were always characterized by the inequality $\bar{K}_{\text{intra}} > \bar{K} > \bar{K}_{\text{inter}}$.

Finally, we call the readers' attention to the standard errors of the various Kappas, which remain stable around 0. We show standard errors to emphasize the statistical nature of the Kappas. However, due to the pre-defined homogeneous connectivity patterns used in our computational experiments, variance (standard deviation) in connectivity is relatively low. Additionally, a small standard deviation is then scaled by a relatively large denominator ($\sqrt{200}$), which reduces it even more.

4.4 An Example of Formal Statistical Testing for Kappas

As discussed previously, one of the strengths of our measures is their statistical definition. This definition allows us to perform formal statistical testing to confirm our conclusions. Here, we illustrate our claim by showing two examples, in Table 7. Our null hypotheses are, in the first test, $\bar{K}_{\text{intra}} \leq \bar{K}$ and, in the second test, $\bar{K}_{\text{inter}} \geq \bar{K}$. The goal of these tests is to formally verify the quality of the clustering identified by an algorithm. If the clustering is good, the null hypotheses $\bar{K}_{\text{intra}} \leq \bar{K}$ and $\bar{K}_{\text{intra}} \geq \bar{K}$ should be rejected, at the usual confidence levels (0.01, 0.05). If the clustering is bad, as it is in our first example, we expect the null not to be rejected.

Table 7. Hypothesis test example

	Test $\bar{K}_{\text{intra}}$	Test $\bar{K}_{\text{inter}}$
Null Hyp	$\bar{K}_{\text{intra}} \leq \bar{K}$	$\bar{K}_{\text{intra}} \geq \bar{K}$
Alt. Hyp	$\bar{K}_{\text{intra}} > \bar{K}$	$\bar{K}_{\text{inter}} < \bar{K}$
Pct inter (actual)	1	0.75
Pct intra (actual)	0.75	1
$\lvert C \rvert$	200	200
$\bar{K}$	1.00	0.56
$\bar{K}_{\text{intra}}$	0.74	na
Std Error	0.01	na
$\bar{K}_{\text{inter}}$	na	0.54
Std Error	na	0.001
t-statistic	-26	-20
Deg freedom	199	19,899
p-value	0.000	0.000
Reject null?	**NO**	**YES**

5 Conclusion

We described a new set of statistical clustering measures that allow formal quality assessments and comparison of algorithms. Our measures are shown to be more robust than the commonly used modularity and conductance. In particular, our measures appear to be more responsive to cluster labeling and less sensitive to sample size, resolution limit and breakdowns during numerical implementation. This latter feature is especially important in the context of larger data sets.

In this article, we restricted our attention to non-overlapping clusters, since that is what most clustering techniques identify. Future investigations could focus on extensions to measuring the strength of overlapping clusters.

Acknowledgements. PM thanks Liudmila Ostroumova Prokhorenkova, Mark Newman, Cris Moore, Aaron Clauset and Anne Morvan, for their helpful comments and guidance. PM was supported by Mitacs-Accelerate PhD award IT05806.

References

1. Almeida, H.M., Guedes, D.O., Meira Jr., W., Zaki, M.J.: Is there a best quality metric for graph clusters? In: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2011, Athens, Greece, 5–9 September 2011, Proceedings, Part I, pp. 44–59 (2011)

2. Aloise, D., Caporossi, G., Hansen, P., Liberti, L., Perron, S., Ruiz, M.: Modularity maximization in networks by variable neighborhood search. In: Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D. (eds.) Graph Partitioning and Graph Clustering, 10th DIMACS Implementation Challenge Workshop, Georgia Institute of Technology, Atlanta, GA, USA, 13–14 February 2012, Proceedings, pp. 113–128 (2012). http://www.ams.org/books/conm/588/11705
3. Biswas, A., Biswas, B.: Defining quality metrics for graph clustering evaluation. Expert Syst. Appl. **71**, 1–17 (2017). http://www.sciencedirect.com/science/article/pii/S0957417416306339
4. Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z., Wagner, D.: On modularity clustering. IEEE Trans. Knowl. Data Eng. **20**(2), 172–188 (2008). https://doi.org/10.1109/TKDE.2007.190689
5. Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Preprint **70**(6), 066111 (2004)
6. Creusefond, J., Largillier, T., Peyronnet, S.: On the evaluation potential of quality functions in community detection for different contexts. ArXiv e-prints, October 2015
7. Djidjev, H., Onus, M.: Using graph partitioning for efficient network modularity optimization. In: Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D. (eds.) Graph Partitioning and Graph Clustering, 10th DIMACS Implementation Challenge Workshop, Georgia Institute of Technology, Atlanta, GA, USA, 13–14 February 2012, Proceedings, pp. 103–112 (2012). http://www.ams.org/books/conm/588/11713
8. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**, 75–174 (2010)
9. Fortunato, S., Barthélemy, M.: Resolution limit in community detection. Proc. Nat. Acad. Sci. **104**(1), 36–41 (2007). http://www.pnas.org/content/104/1/36.abstract
10. Holder, L.B., Caceres, R., Gleich, D.F., Riedy, J., Khan, M., Chawla, N.V., Kumar, R., Wu, Y., Klymko, C., Eliassi-Rad, T., Prakash, A.: Current and future challenges in mining large networks: report on the second SDM workshop on mining networks and graphs. SIGKDD Explor. Newsl. **18**(1), 39–45 (2016). http://doi.acm.org/10.1145/2980765.2980770
11. Huang, H., Liu, Y., Hayes, D., Nobel, A., Marron, J., Hennig, C.: (15) Significance testing in clustering. In: Hennig, C., Meila, M., Murtagh, F., Rocci, R. (eds.) Handbook of Cluster Analysis, pp. 315–335. Chapman and Hall/CRC (2015)
12. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **78**, 046110 (2008)
13. Leskovec, J., Lang, K.J., Mahoney, M.W.: Empirical comparison of algorithms for network community detection. ArXiv e-prints, April 2010
14. Leskovec, J., Lang, K.J., Dasgupta, A., Mahoney, M.W.: Statistical properties of community structure in large social and information networks. In: 7th International Conference on WWW (2008)
15. Morvan, A., Choromanski, K., Gouy-Pailler, C., Atif, J.: Graph sketching-based massive data clustering. In: SIAM International Conference on Data Mining (SDM 2018) (2018, to appear)
16. Moschopoulos, C.N., Pavlopoulos, G.A., Iacucci, E., Aerts, J., Likothanassis, S., Schneider, R., Kossida, S.: Which clustering algorithm is better for predicting protein complexes? BMC Res. Notes **4**(1), 549 (2011), https://doi.org/10.1186/1756-0500-4-549
17. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **69**, 026113 (2004)

18. Ostroumova Prokhorenkova, L., Prałat, P., Raigorodskii, A.: Modularity of complex networks models. In: Bonato, A., Graham, F.C., Prałat, P. (eds.) Algorithms and Models for the Web Graph, pp. 115–126. Springer, Cham (2016)
19. Ostroumova Prokhorenkova, L., Prałat, P., Raigorodskii, A.: Modularity in several random graph models. Electron. Notes Discrete Math. **61**, 947–953 (2017). http://www.sciencedirect.com/science/article/pii/S1571065317302238. The European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB 2017)
20. Reichardt, J., Bornholdt, S.: When are networks truly modular? Physica D Nonlinear Phenom. **224**(1), 20–26 (2006). http://www.sciencedirect.com/science/article/pii/S0167278906003678. Dynamics on Complex Networks and Applications
21. Sanders, P., Schulz, C.: High quality graph partitioning. In: Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D. (eds.) Graph Partitioning and Graph Clustering, 10th DIMACS Implementation Challenge Workshop, Georgia Institute of Technology, Atlanta, GA, USA, 13–14 February 2012, Proceedings, pp. 1–18 (2012). http://www.ams.org/books/conm/588/11700
22. Spielman, D.A., Teng, S.H.: A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning. SIAM J. Comput. **42**(1), 1–26 (2013)
23. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. CoRR abs/1205.6233 (2012). http://arxiv.org/abs/1205.6233
24. Yang, J., Leskovec, J.: Overlapping community detection at scale: a nonnegative matrix factorization approach. In: WSDM 2013. ACM, 978-1-4503-1869-3/13/02 (2013)